REMARKS/ARGUMENTS

Reconsideration of this application is respectfully requested. The following remarks are responsive to the Office Action mailed March 25, 2004. Applicants appreciate the Examiner Hoang's time and consideration in a telephonic interview of July 1, 2004. Independent claims 1, 9, 17, and 18 have been amended based upon discussions from the interview regarding possible clarifying amendments with respect to the teachings of the prior art.

Claims 1-6, 8-14, and 16-18 are pending.

Claims 1-6, 8, 17 and 18 stand rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over Dzikewich et al., U.S. Patent No. 5,706,500 ("Dzikewich") in view of Horiguchi et al., U.S. Patent No. 6,073,157 ("Horiguchi").

Claims 9-14, and 16 stand rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over Dzikewich in view Horiguchi in further view of Matsuda et al., U.S. Patent No. 5,790,419 ("Matsuda").

To establish a **prima facie** case of **obviousness**, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. In re Vaeck, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Claims 1-6, 8, 17 and 18 are not rendered obvious under 35 U.S.C. §103(a) in view of Dzikewich and Horiguchi.

The Office Action has rejected claims 1-6, 8, 17 and 18 under 35 U.S.C. §103(a) as allegedly being unpatentable over Dzikewich in view Horiguchi. For the present claims to be

-7-

rendered obvious by Dzikewich in view of Horiguchi, the cited art must individually or in combination teach each and every feature of the present claims. Furthermore, there must be some motivation or suggestion to combine the cited art.

Claim 1 includes the following limitations:

automatically detecting an exit of a child application object launched by a parent application object, <u>wherein the computer system which maintains the hierarchy of application objects, the parent application object, and a grandchild application object, launched by the child application object, remain active;</u>

automatically terminating the grandchild application object after the exit of the child application object;

automatically determining whether the exit of the child application object was expected;

automatically attempting restart of the child application object if the exit of the child application object was unexpected; and

automatically signaling an outcome of the attempted restart of the child application object to the parent application object <u>that launched the child application object prior to the exit of the child application object.</u>

(Claim 1, emphasis added)

The teachings of Dzikewich in combination with Horiguchi fails to render the present claims obvious because neither individually or in combination teach or suggest automatically detecting an exit of a child application object launched by a parent application object, <u>wherein the computer system which maintains the hierarchy of application objects, the parent application object, and a grandchild application object, launched by the child application object, remain</u>

-8-

active. Dzikewich teaches the operation of a business event processor for processing at least one unit of work. <u>Each unit of work is data</u> that includes an application context and a script processing state. The script processing state indicates which unit of work the business event processor has already processed. The inclusion of the application context and the script processing state allows the business event processor to restart where it left off after a process failure <u>resulting from a system failure</u>. Only after <u>a system failure is the business event processor restarted</u> and the step level recovery script (restart script) automatically run and the units of work (data) loaded and processed to begin the operations at a point where the system failure occurred.

In other words, in Dzikewich <u>the system has to completely fail or completely restart</u> for the system level recovery script to run in order to <u>detect a failure</u> and to determine whether or not that failure was expected. In contrast, the present claims recite <u>automatically detecting an exit of a child application</u> object launched by a parent application object, wherein <u>the computer system</u> which maintains the hierarchy of application objects, <u>the parent application object, and a grandchild application object</u>, launched by the child application object, <u>remain active</u> and automatically <u>determine whether the exit of the child application object was expected.</u> Neither the parent application nor the computer system need fail and be restarted to detect the exit of the child application or to determine whether or not the exit was expected.

As indicated in the present Office Action, Dzikewich does not teach automatically terminating the grandchild application object after the exit of the child application object.

Adding what is taught in Horiguchi fails to cure the deficiencies of Dzikewich as detailed above. Horiguchi does not teach any of the elements or features recited in claim 1. Horiguchi

-9-

discusses the execution of processes where each may comprise of one or more enclaves. An enclave is defined as a logical run-time structure that supports the execution of a group of procedures and can have multiple threads. A thread is an execution construct consisting of synchronous invocations and terminations of invocation units (including procedures). In other words, the hierarchy of levels is as follows: at the top is processes, which in turn are made up of enclaves, which are in turn made up of threads which can be a group of procedures. It is important to note, however, that there is no hierarchal relationship among processes (column 3, ll. 34-35) and no hierarchal relationship among threads (column 7, ll. 7-9). Because Horiguchi only discusses processes, enclaves, and threads, and does not discuss the limitations as recited in claim 1 of the present invention, claim 1 is patentable over the cited art. For at least these reasons, the dependent claims 2-8 are also patentable over the cited art.

Because independent claims 9, 17 and 18 have substantially similar limitations as claim 1, the same arguments that applied to claim 1 also apply to claims 9, 17 and 18. Therefore, for at least the reasons stated above, independent claims 1, 9, 17 and 18 and the dependent claim limitations therefrom are patentable over the cited art.

The teachings of Matsuda fail to cure the deficiencies of Dzikewich and Horiguchi. Matsuda discloses a watchdog timer that monitors a pulse train frequency from a controller within the microcomputer circuit. When the pulse train falls outside a specified frequency range, the microcomputer is in an abnormal state and the watchdog timer outputs a signal to the controller that causes the microcomputer to reset. Matsuda merely detects an error in the microcomputer's state based on a pulse train frequency measurement and not an exit of the child

-10-

application object. The microcomputer itself does not exit, rather it only outputs an erroneous pulse train. Although Matsuda may discuss the use of a watchdog, it does not discuss or teach any of the other claim limitations of the independent claims and therefore does not provide what is missing from the teachings of Dzikewich and Horiguchi.

Because Matsuda does not cure the deficiencies of Dzikewich and Horiguchi, the independent claims 1, 9, 17, and 18 and the dependent claims limitations therefrom are patentable over the cited art.

Authorization is hereby given to charge our Deposit Account No. 02-2666 for any charges that may be due. Furthermore, if an extension is required, then Applicants hereby request such an extension.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: 7/21, 2004

Mark R. Vatuone
Reg. No.: 53,719

12400 Wilshire Blvd.
Seventh Floor
Los Angeles, CA 90025
(408) 947-8200

-11-